Remarks

Claims 1-20 are pending in this application. The examiner has rejected claims 1-11 and 13-20 as being obvious under 35 U.S.C. 103(a) over "Kernel Korner Writing a Linux Driver" by Matia in view of "SCONE: Using Concurrent Objects for Low-level Operating System Programming" by Itoh. The examiner has additionally rejected claim 12 under 35 U.S.C. 103(a) over Matia in view of Itoh and U.S. Patent No. 6,754,858 to Broman.

A. The Combination of Matia and Itoh Does Not Establish a Prima Facie Case of Obviousness as to Independent Claims 1, 8, and 13

Applicants respectfully submit that a prima facie case of obviousness has not been established and that a rejection of the pending claims on obviousness grounds is improper. A prima facie case of obviousness requires a showing that all of the claim limitations of the rejected claims are taught or suggested by the prior art. Manual of Patent Examining Procedure 2143 and 2143.03. The establishment of a prima facie case of obviousness requires that *all* the claim limitations be taught or suggested by the prior art. MPEP 2143.01 (emphasis added). "All words of a claim must be considered in judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 1385, 165 U.S.P.Q. 494, 496 (CCPA 1970). Here, a prima face case of obviousness is not established because (a) the combination of Matia and Itoh does not disclose or suggest "a device driver having . . . a service layer in open source form"; and (b) the combination of Matia and Itoh does not disclose or suggest the step of "compiling the service layer against the kernel . . . after each modification to the kernel."

1. The Combination of Matia and Itoh Does Not Disclose or Suggest a Device Driver with a Service Layer in Open Source Form

According to the Examiner, Matia does not explicitly teach the device driver having a service layer that interfaces between the kernel of the operating system and the at least one executable module of the device driver. (Office Action, p.3) Thus, Matia does not teach or suggest a device driver with a service layer in open source form. The Examiner goes on to state that Itoh teaches a device driver having a service layer for each type of low-level system code used for communicating with kernel and device. (Office Action, p.3) The cited portions of Itoh describe providing dedicated system service layers for each type of low-level system code so that the "primitive operations of concurrent objects are mapped onto specific functionality of low-level system services." (Itoh, p.388) The cited portions of Itoh, however, make no mention or suggestion of service layers in **open source form**. This disclosure is absent from the cited passages of Itoh. Because Itoh does not disclose a service layer in open source form, all of the claim limitations are not taught or suggested by the prior art. As such, for at least this reason, the Examiner's obviousness rejection of claims 1, 8, and 13 should be withdrawn.

2. The Combination of Matia and Itoh Does Not Disclose or Suggest Compiling the Driver against the Kernel after Each Modification to the Kernel

According to the Examiner, Matia teaches the step of "compiling the driver against the kernel of the open source operating system after each modification to the kernel of the open source operating system." (Office Action, p. 3). The Examiner explains that this teaching is found on pages 2, 10, and 11 of Matia. Page 2 of Matia describes drivers generally, but does not concern driver compilation. Page 10 concerns the "task of integrating the driver into the kernel" and describes the step of "re-compile the driver." The recompilation of Matia, however, occurs after a modification to the **driver**, and **not** after each modification to the **kernel**, as required by the independent claims of the application.

Matia concerns the recompilation of the device driver following a modification to the device driver. As an example, on page 8 of Matia, under the heading "Implementation of Driver Functions, the user is given instructions on "programming your own driver." These instructions continue through page 10 and concern steps for recompiling the driver following a modification *to the driver itself.* Additionally, on page 11 of Matia, the user is told that it is recommended that the driver be compiled **alone** before linking the kernel. This is not the same as compiling the service layer **against the kernel,** as required by the independent claims. Additionally, Matia, on page 11, describes configuring the kernel after compiling the driver alone. Matia does not disclose the element of the independent claims that requires that the server layer be compiled against the kernel **after** each modification to the kernel. The Examiner does not refer to Itoh as disclosing or suggesting these elements. Because these elements of the claims are not taught or suggested by Matia in combination with Itoh, a prima facie case of obviousness cannot be established by the combination of Matia and Itoh. As such, the rejection of claims 1, 8, and 13 should be withdrawn.

B. Dependent Claims 1-7, 9-12, and 14-20

Dependent claims 1-7, 9-12, and 14-20 will not be discussed individually herein, as these claims depend, either directly or indirectly, from an otherwise allowable base claim.

Conclusion

Applicants respectfully submit that the pending claims 1-20 of the present invention, as previously amended, are allowable. Applicants respectfully request that the rejection of the pending claims be withdrawn and that these claims be passed to issuance.

Respectfully submitted,

Roger Fulghum
Registration No. 39,678

Baker Botts L.L.P.
910 Louisiana
One Shell Plaza
Houston, Texas 77002-4995
(713) 229-1707

Baker Botts Docket Number: 016295.0732

Date: November 22, 2006